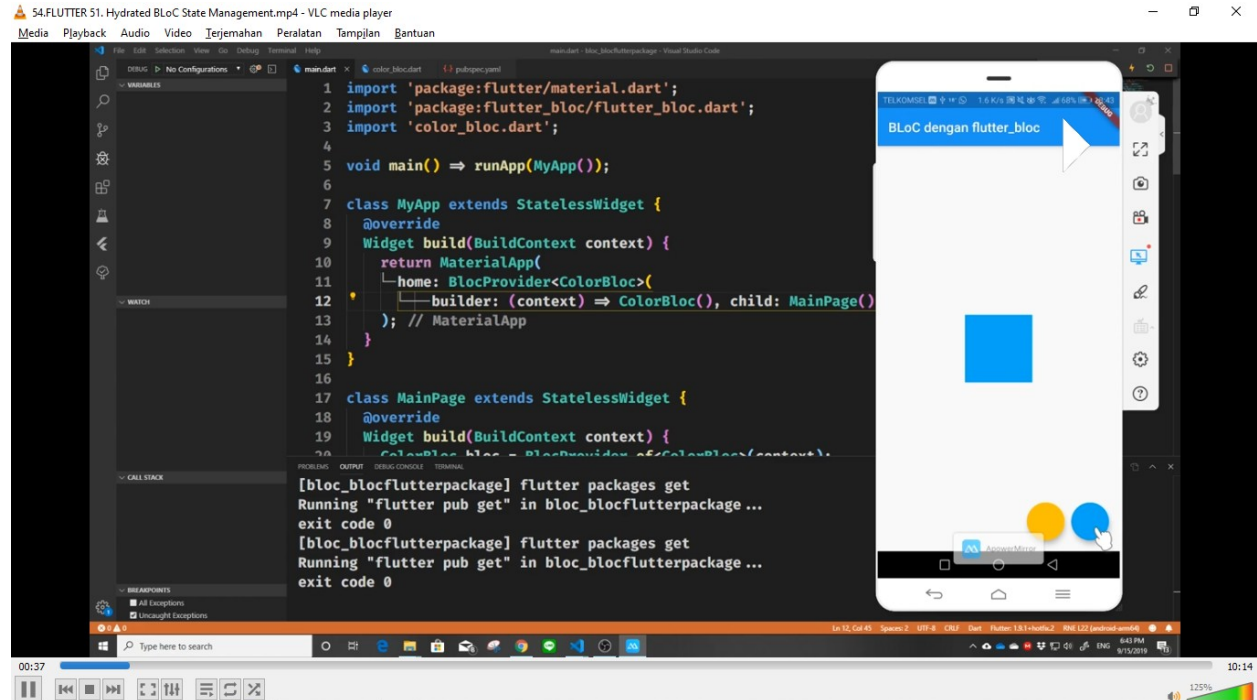


Pada kali ini pada tutorial 45 pernah menjelaskan blocState management
Pada tutorial tersebut buat aplikasi seperti ini

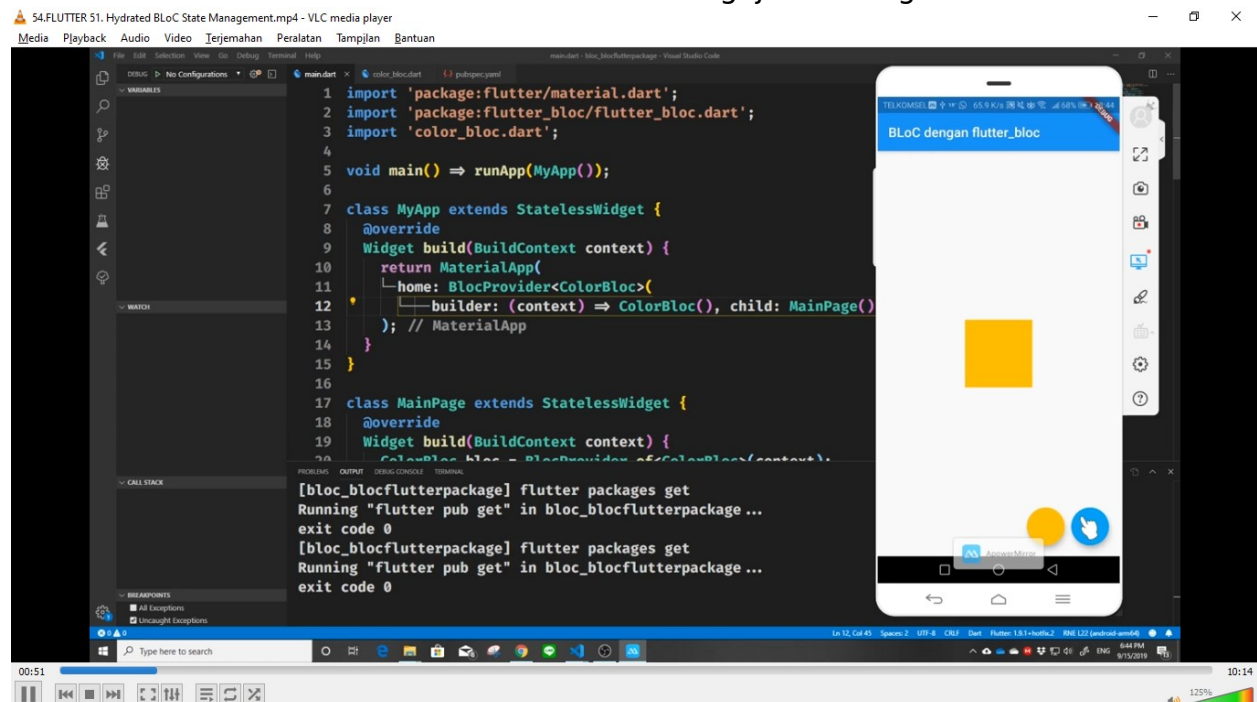


Ada animatedContainer dan 2 tombol ketika di tekan maka akan berubah warna
animatedContainer

Sekarang permasalahannya adalah

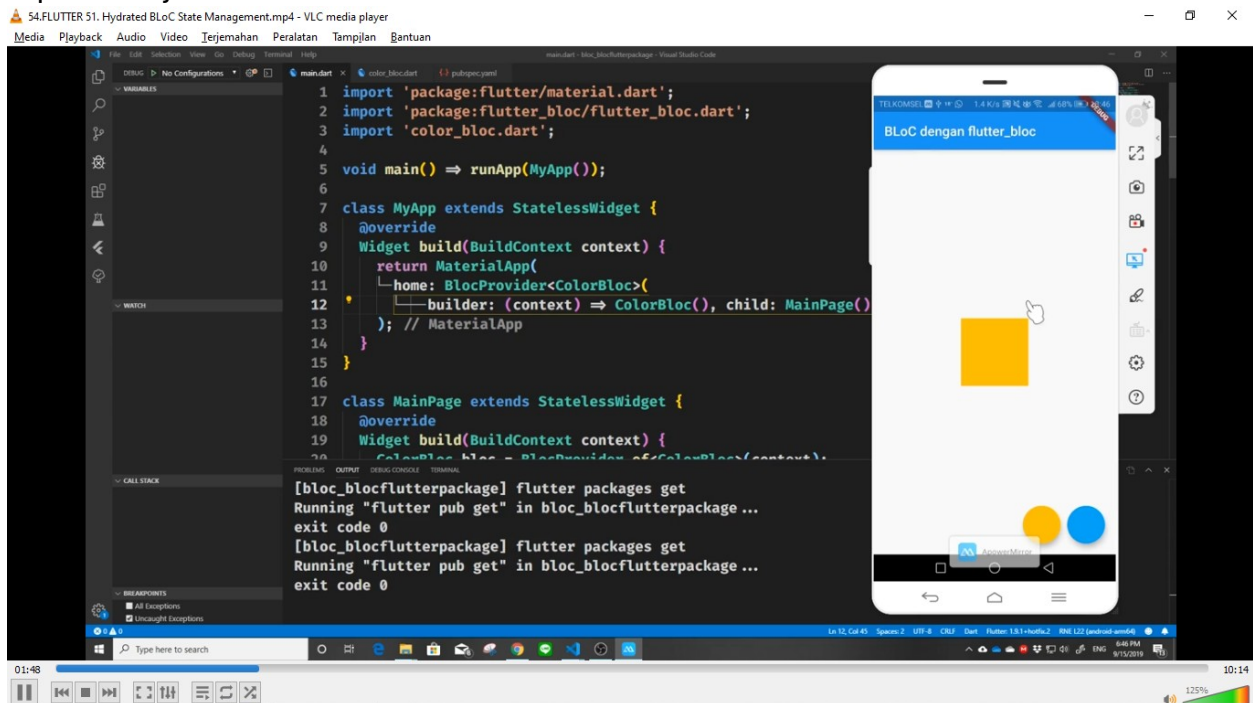
Ketika di close aplikasi ini lalu buka lagi

State terahir kali itu warna biru ketika di buka lagi jadi kuning

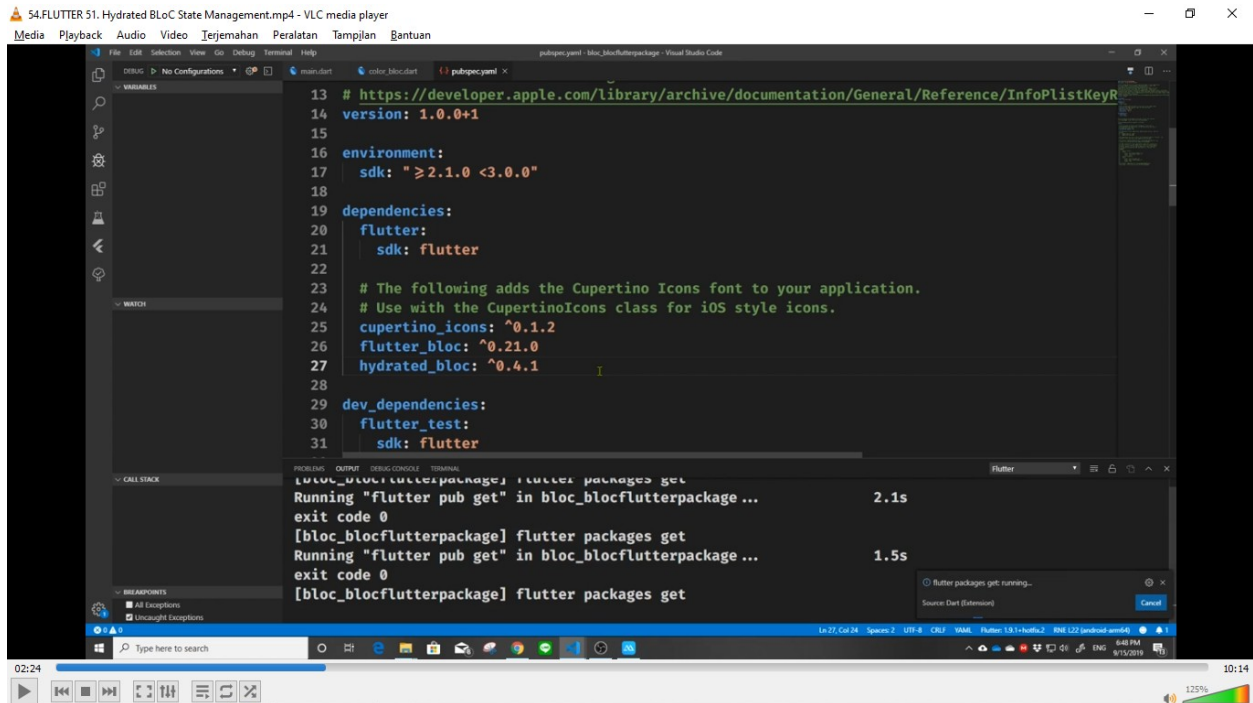


Dia kembali ke awal yaitu warna ember

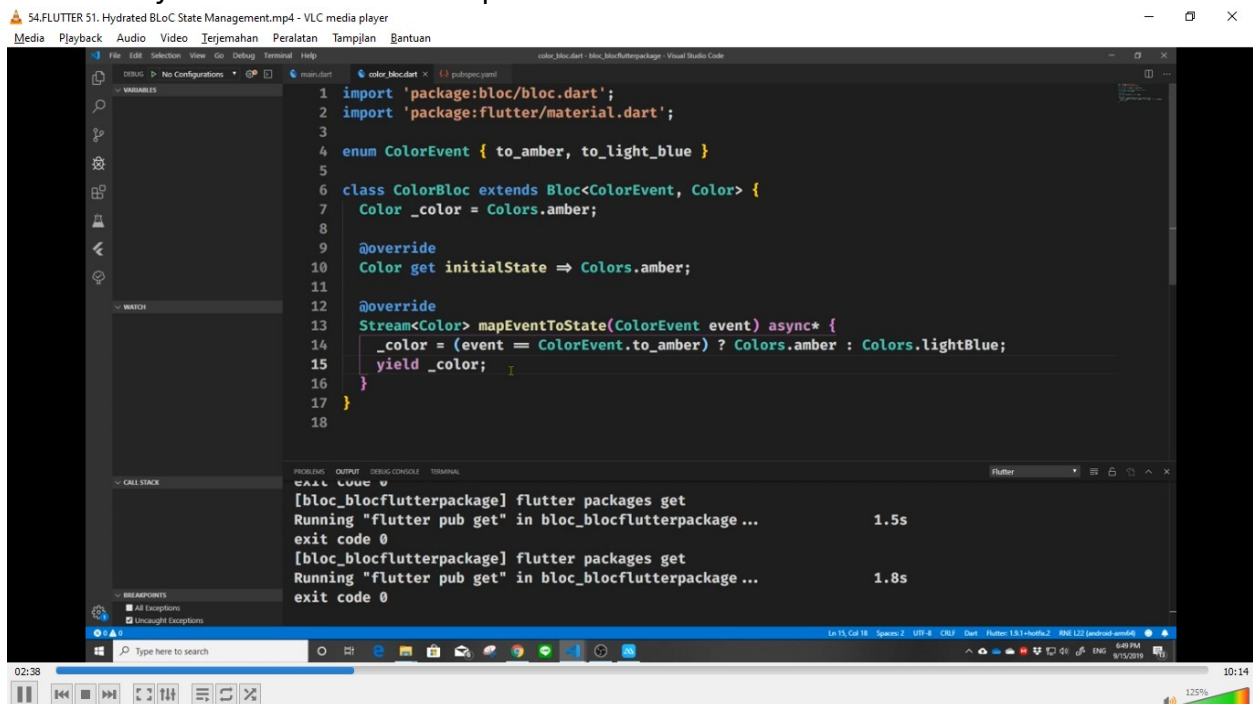
Pada kali ini menjelaskan menggunakan hydrated Bloc
hydrated Bloc ini ssebernya blocState management tapi versi sudah di uprade
hydrated Bloc dia dapat menyimpan state terakhir kita otomatis
ketika buka dia akan load lagi state terakhir
misalkan state terakhir kali di buka warna biru
dan buka lagi maka menjadi biru
pada kali ini tidak buat dari 0
tapi melanjutkan tutorial ini



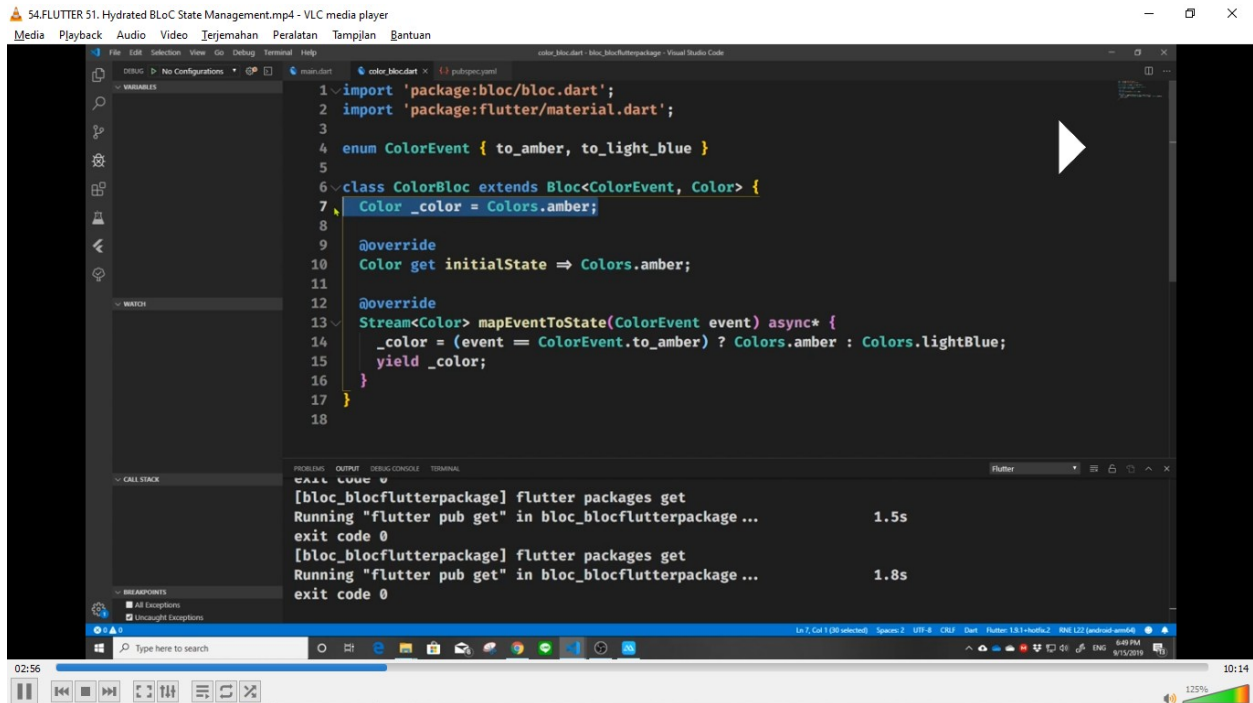
Pertama ke pubspec yml dulu
Gunkan flutter bloc 20 kita ubah menjadi 21.0
Lalu tambhagn lagi package hydrated_bloc



Kalau sudah masuk ke class color_bloc
Sebelumnya sudah membuat seperti ini



Sebelum nya lanjut menjelaskan hydrated bloc
Ingin memberitahukan menggunakan bloc dart kita tidak perlu lagi
Memiliki satu buat yield untuk menyimpan state kita contoh yg color ini tidak perlu
lagi disimpan



Kita hapus

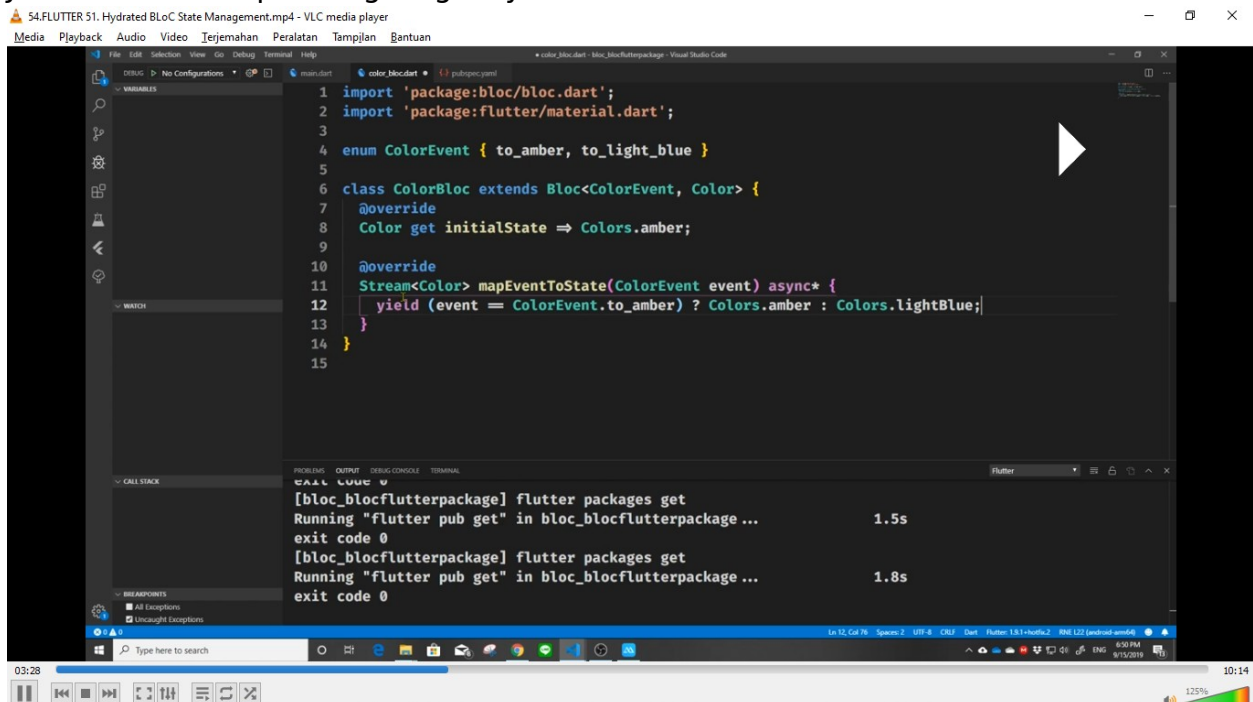
Dia akan disimpan secara otomatis oleh state didalam class bloc

Nanti akan diturunkan clas kita kedalam colors bloc

Colors bloc ini kita inherinte dari clas bloc

Disimpan dibloc itu ketika diupdate ketika diberikan stream

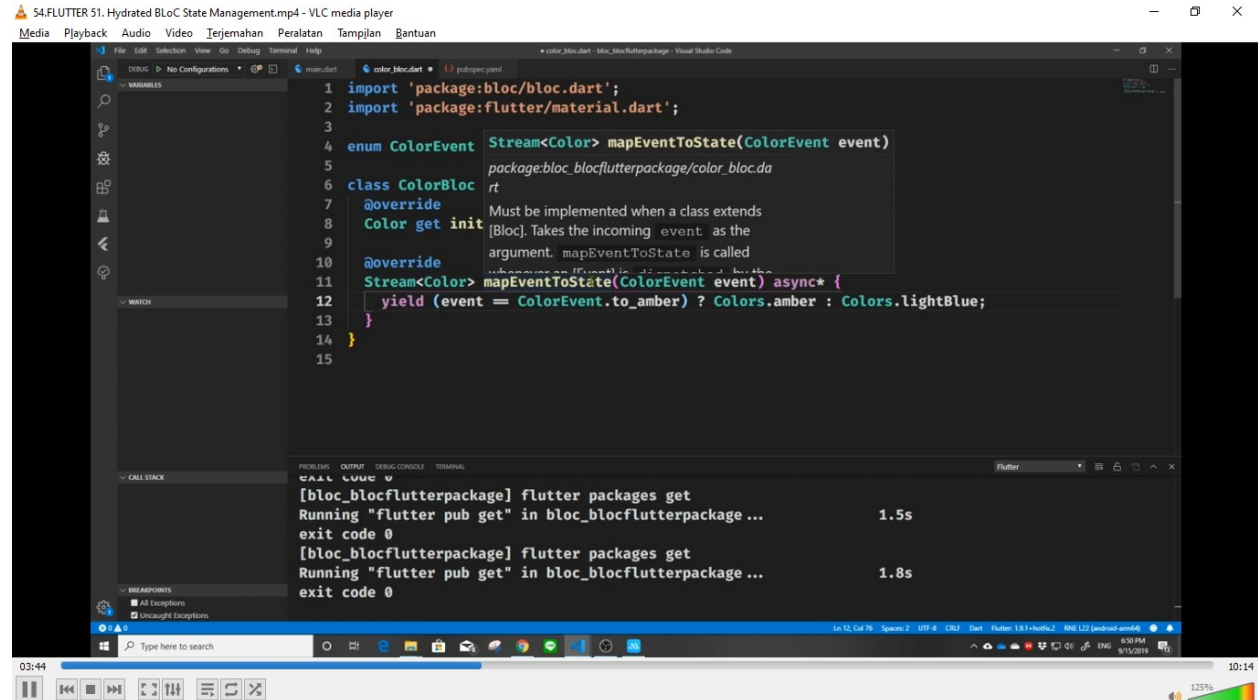
Jadi disini kita hapus langsung ke yield



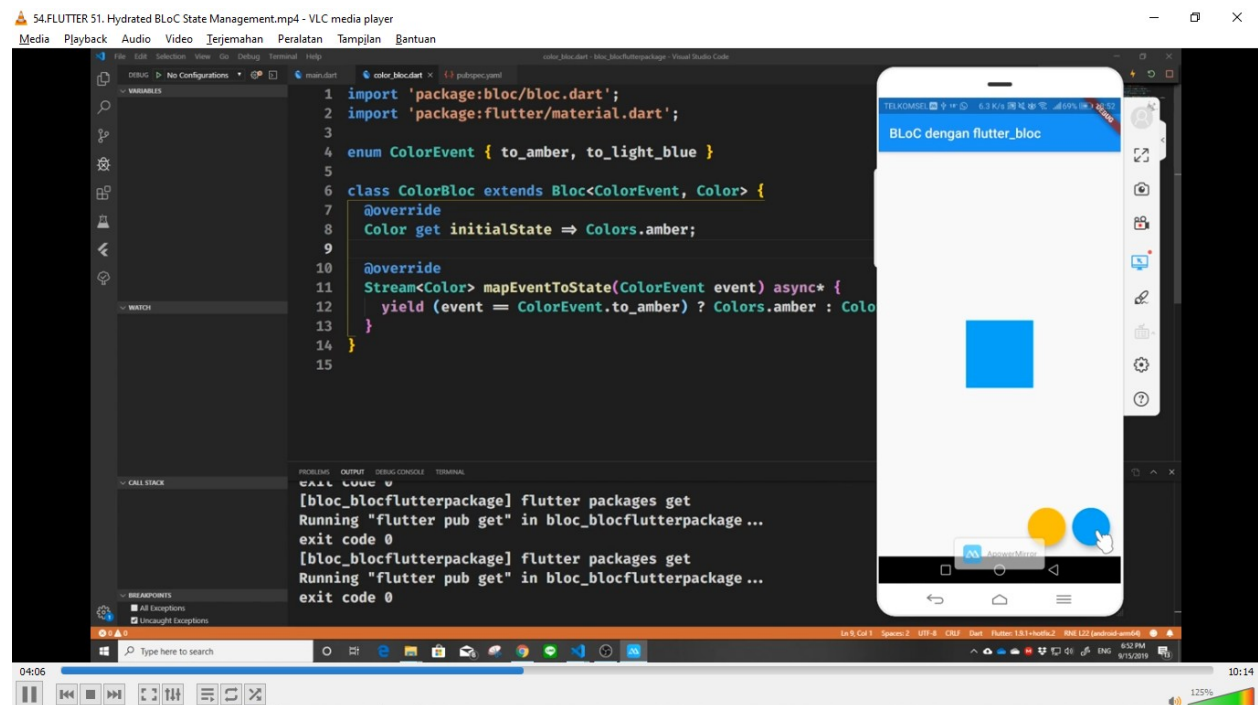
Jadi maka akan kembalikan kedalam stream yg ini

Selain kembalikan kedalam stream ini akan di update

State yg disimpan class dari bloc kita
Bisa dilihat keterangan nya disini

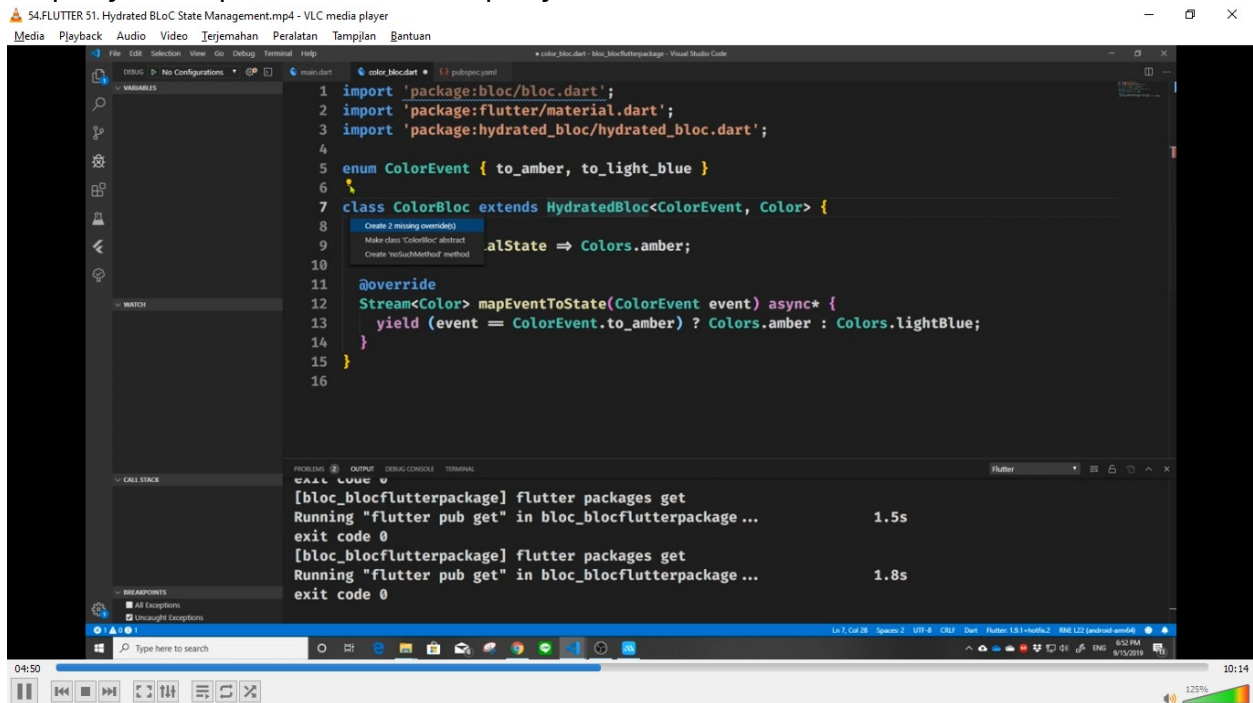


Jadi nanti ada new state yg ini
Kita coba dulu run kan
Ini ditekan

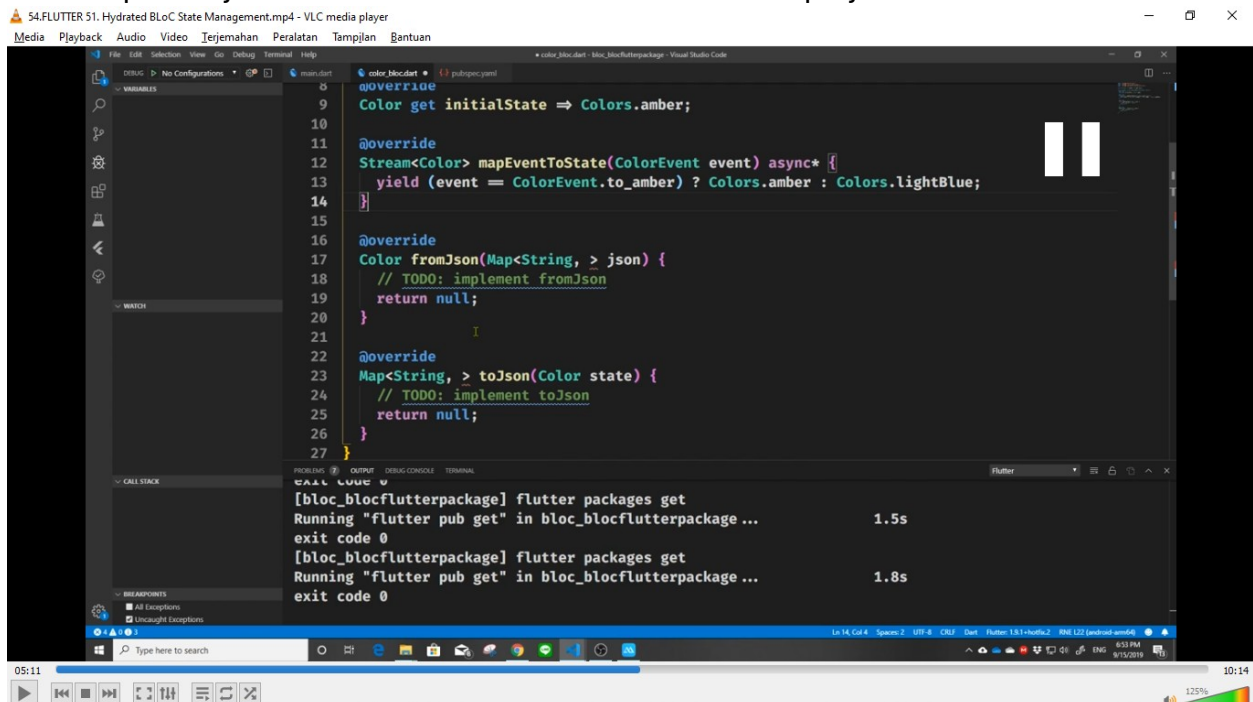


Maka sama saja gk perlu lagi disimpan sebuah yield
Kembali lagi pertama2 kita lakukan adalah mengubah bloc ini menjadi hydrated bloc

Jadi gk gk extends ke bloc lagi tapi hydrated bloc
Kita import dahulu
hydrated bloc
ini punya berapa method masih punya untuk kita overread



Crete2
Ada fromJson dan toJson
fromJson ini akan dipakai ketika aplikasi kita dijalankan
ketika aplikasi jalankan diakan load data terakhir berupa json

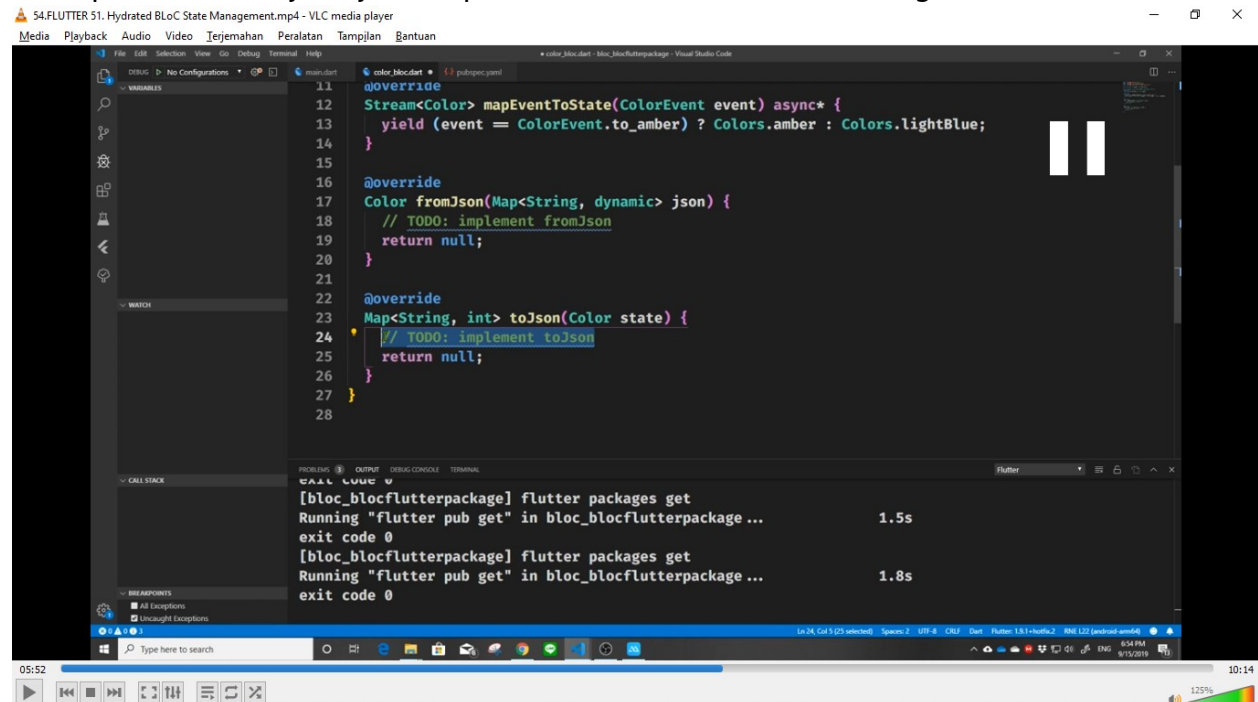


Ketika kita mau melakukan perubahan terhadap state kita dia akan save lagi bentuk json dia akan memanggil method toJson

Sekarang kita pasang

Tipe map nya string to dynamic

Lalu save di toJson maka map String to integer bukan colors karena color tidak bisa disimpan kedalam json jadi terpaksa kita ubah kedalam integer dahulu



Pertama kita implemen dulu toJson

Maka disini

Buat try catch

Di try kita return kembalikan map string to integer

Kita kasih key terserah diberikan kalau ini color : jadi si state itu adalah color sekarang yg digunakan yaitu ember ata blue

Ini tidak bisa disave kedalam bentuk json maka kita ubah jadi integer .value dia akan mengunah integer

Sekarang ke fromJson

Try catch

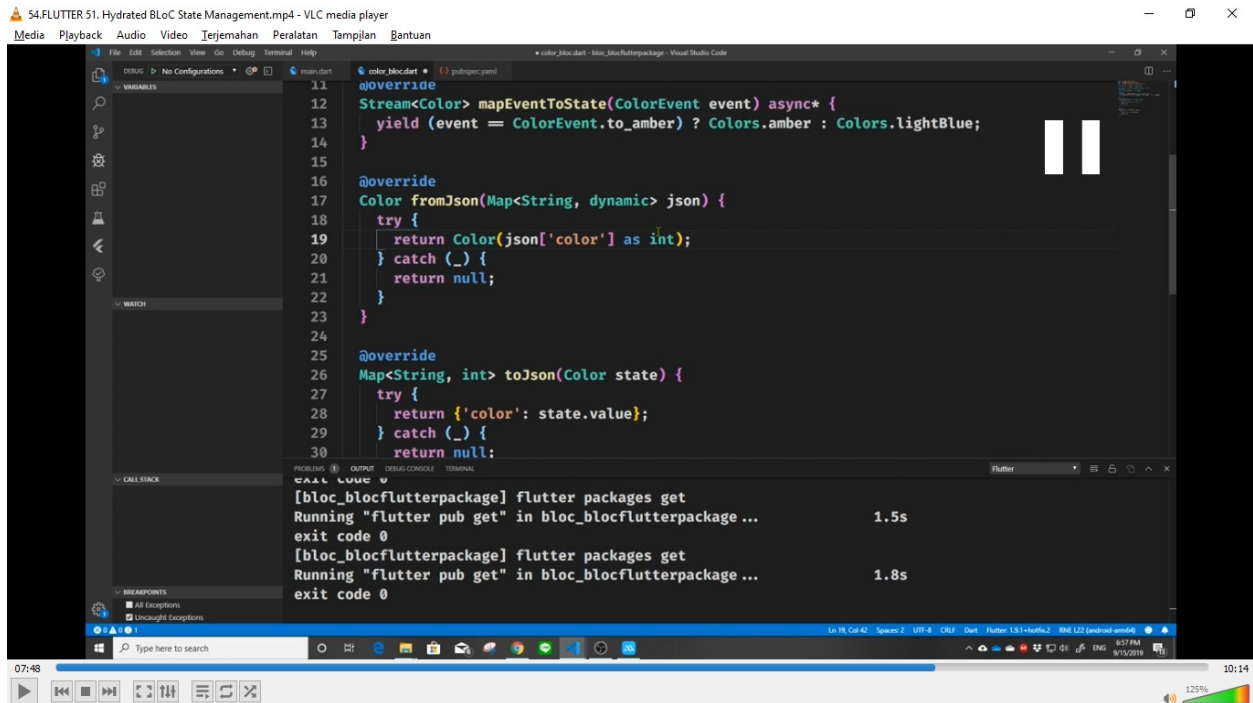
Kalau dia gagal atau tidak ada save maka akan kembalikan null

Kalau ada dia akan kemablikan kedalam json

Kasih value nya color sebenarnya intger oleh krena itu buat as int

Untuk ubah int ke colors lagi itu mudah tinggal

Berikan color() lalu masukan value untuk ubah integer ke color kembali



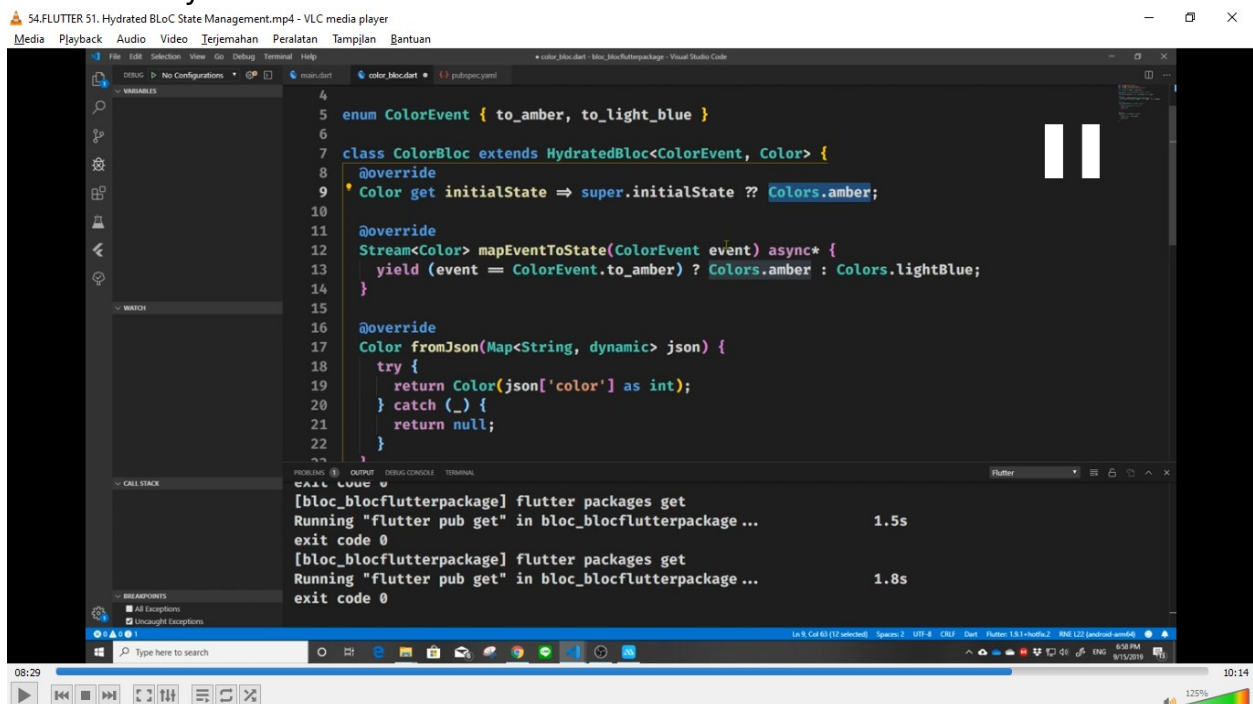
Lalu ke atas

Di initialState kita perlu cek dulu apakah ada save sebelumnya

Maka buat super.initial state tanda tanya dua kali

Maksud nya apaka untuk cek apakah super initial state nul atau tidak kalau tidak dia akan kembalikan ini

Kalau hasilnya null dia akan kembalikan colors. Ember



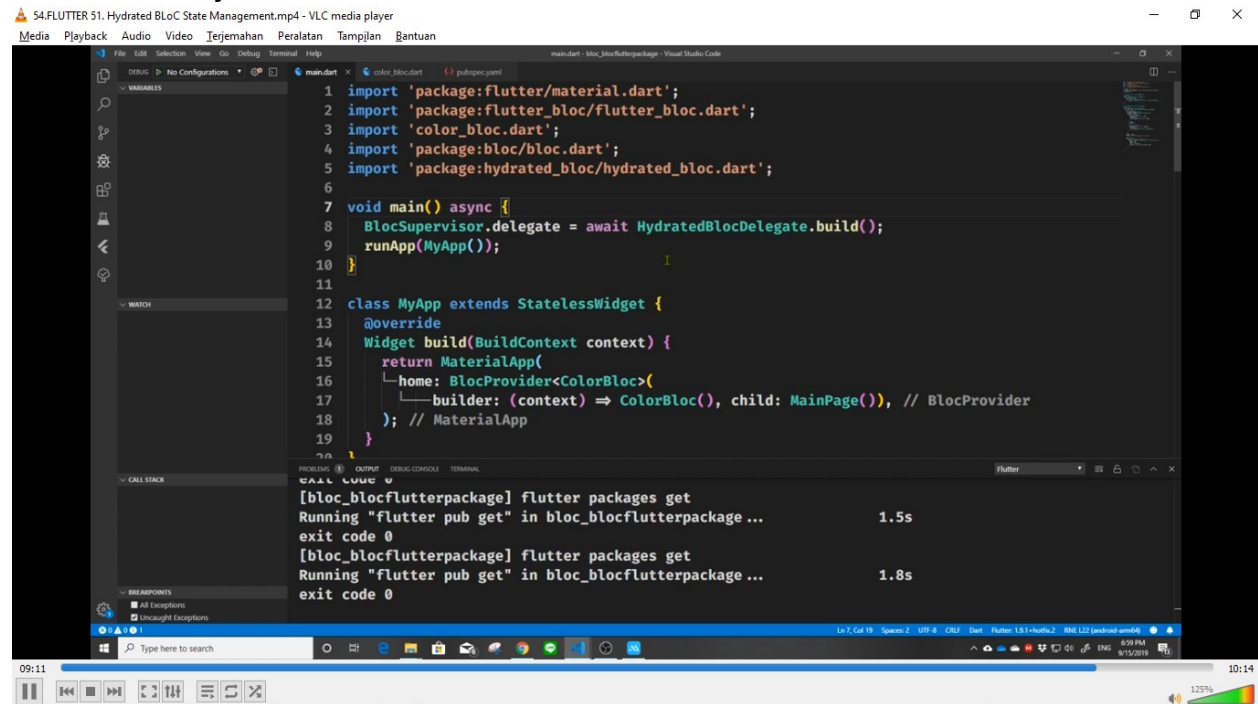
Ke main dart kita perlu ubah sedikit

Kita import dulu bloc dart

Dan satu hyrade bloc

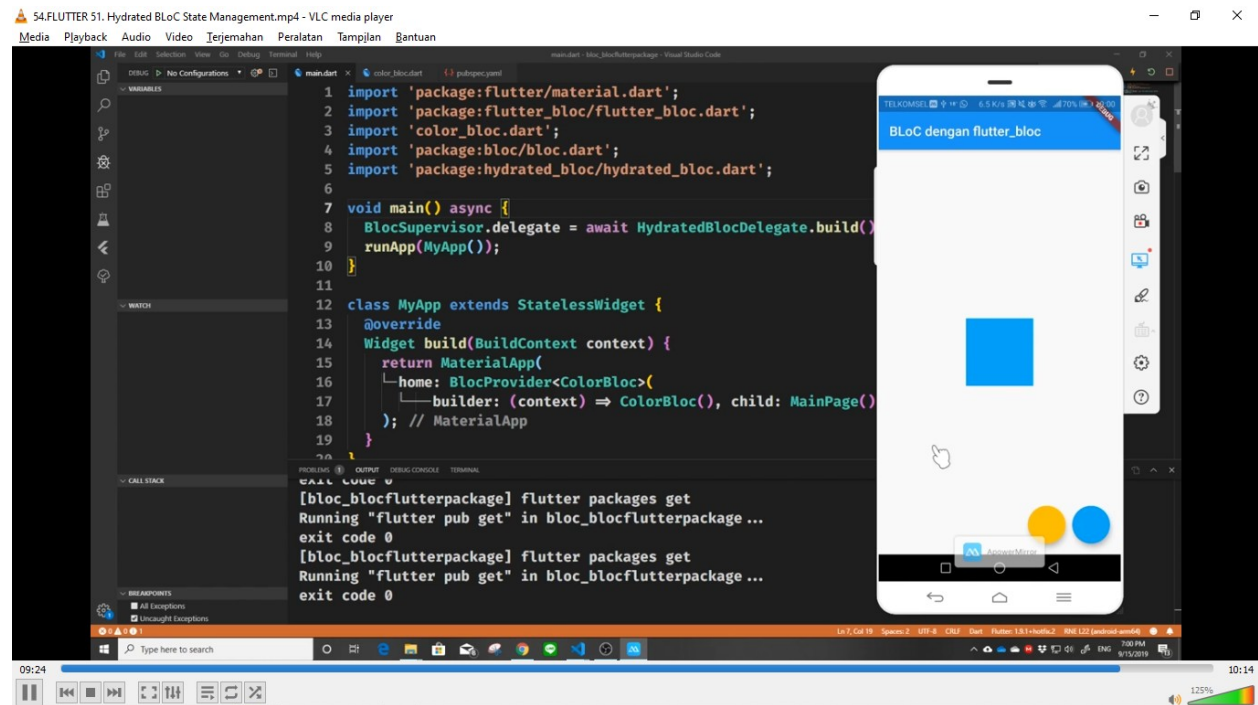
Disini main tingal buat blocSupervisor.delegate = await ganti dgn hy.build;

Disni kasih async

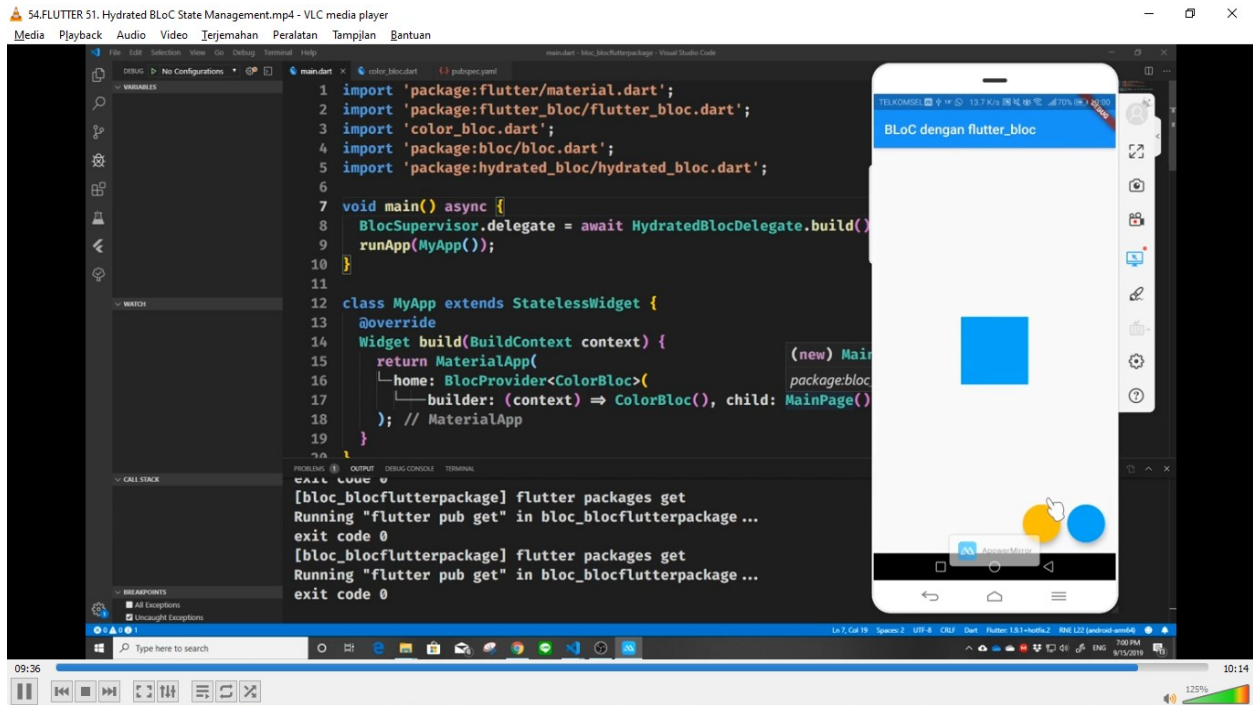


lalu jalankan

Terahir warna biru



Dan coba close lalu buka lagi



Tetap biru

Seakrang bisa save satet kita terakhir

Sbarnya untuk save seperti ini kita mengunkan bloc biasa lalu digabung shared_p
Cuman pakai hydrated bloc ini semua menjadi otomatis memudah kan kita